AMENDMENTS TO THE CLAIMS

Please amend claims 1 and 7, cancel claims 2 – 4 and 6, and add new claims 12 – 17, as set forth in the listing of claims that follows:

Listing of Claims:

(insert listing of amended/canceled claims)

1. (Currently Amended) A method for generating multiple high-resolution pulse width modulated (PWM) signals comprising the steps of:

receiving data representative of duty cycle values;

sorting said duty cycle values in a duty cycle table to generate a PWM generation table;

generating an interrupt from a capture and compare interrupt defined by a match between a timer value and a duty cycle register;

generating an interrupt from a timer overflow;

generating multiple PWM signals as defined by said PWM generation table upon detection of said capture and compare interrupt; and

returning to a beginning point in said PWM generation table upon generation of a timer overflow interrupt,

wherein said step of sorting said duty cycle values further comprises the steps of:

sorting said duty cycle values along with data stored in a port table containing port pin assignments for each duty cycle value; and

sorting said duty cycle values along with data stored in a bit-mask table containing bit-mask assignments corresponding to a specific port pin assignment.

2. (Canceled)

3. (Canceled)

4. (Canceled)

5. (Original) The method as claimed in claim 1 wherein said step of sorting said duty cycle values is done in a background task.

6. (Canceled)

7. (Currently Amended) A system for generating multiple high-resolution pulse width modulated (PWM) signals comprising:

a microprocessor having a timer, said microprocessor for receiving duty cycle values;

a communication bus for sending duty cycle values to said microprocessor;

a capture and compare module operative to selectively generate software interrupts as a function of timer and duty cycle values in communication with said microprocessor;

a duty cycle table at a predetermined location in memory of said microprocessor, said duty cycle table for storing said duty cycle values;

a port table having port assignments corresponding to a duty cycle value in said duty cycle table, said port table being stored in memory of said microprocessor;

a bit-mask table having bit-mask assignments corresponding to a port assignment in said port table, said bit-mask table being stored in memory of said microprocessor;

a PWM generation table created by sorting said duty cycle table, said port table and said bit-mask table upon receipt of all duty cycle values; and

an interrupt routine that is entered only upon generation of an interrupt wherein said interrupt routine allows said microprocessor to write a predetermined duty cycle value from said PWM generation table to a predetermined port assignment from said PWM generation table.

8. (Original) The system as claimed in claim 7 wherein said interrupt is generated by a match between a value of said timer and a duty cycle value; and

said predetermined values are written to said port assignment from said PWM generation table.

9. (Original) The system as claimed in claim 7 wherein said interrupt is generated by a timer overflow and said predetermined values for said port assignments are all low.

10. (Original) A method for generating multiple high-resolution pulse width modulated (PWM) signals in a system having a microprocessor, a communication bus, and a timer, said method comprising the steps of:

receiving a plurality of duty cycle values at said microprocessor;

placing said duty cycle values in a duty cycle table wherein each duty cycle value has a corresponding port assignment in a port table and a corresponding bit-mask assignment in a bit-mask table, said port and bit-mask tables being embedded in software in said microprocessor;

generating a PWM generation table by sorting said duty cycle, port and bit-mask tables in a background task of said microprocessor;

generating a capture and compare interrupt when a source of said interrupt is when a timer value matches a duty cycle value;

generating a timer overflow interrupt when a source of said interrupt is an overflow of said timer;

receiving an interrupt at said microprocessor;

determining a source for said received interrupt;

writing predetermined values to a plurality of pins on said microprocessor based on said source for said interrupt;

writing values to a plurality of ports wherein said values are taken directly from said PWM generation table during a capture and compare interrupt;

writing a low value to a plurality of pins during a timer overflow interrupt; and

returning to a beginning point in said PWM generation table upon generation of a timer overflow interrupt.

11. (Original) The method as claimed in claim 10 wherein said step of generating a PWM generation table further comprises placing an invalid duty cycle value at the end of said PWM generation table, said invalid duty cycle value being a value that can never be equal to a value of said timer.

12. (New) A method for generating multiple high-resolution pulse width modulated (PWM) signals comprising the steps of:

receiving data representative of duty cycle values;

sorting said duty cycle values in a duty cycle table to generate a PWM generation table;

generating an interrupt from a capture and compare interrupt defined by a match between a timer value and a duty cycle register;

generating an interrupt from a timer overflow;

generating multiple PWM signals as defined by said PWM generation table upon detection of said capture and compare interrupt; and

returning to a beginning point in said PWM generation table upon generation of a timer overflow interrupt,

wherein said step of sorting said duty cycle values to generate a PWM generation table further comprises the step of placing an invalid duty cycle value at the end of the PWM generation table, wherein said invalid duty cycle value is a value that is not equal to said timer value for allowing the timer to overflow.

13. (New) A method for generating multiple high-resolution pulse width modulated (PWM) signals comprising the steps of:

receiving data representative of duty cycle values;

sorting said duty cycle values in a duty cycle table to generate a PWM generation table;

generating an interrupt from a capture and compare interrupt defined by a match between a timer value and a duty cycle register;

generating an interrupt from a timer overflow;

generating multiple PWM signals as defined by said PWM generation table upon detection of said capture and compare interrupt; and

returning to a beginning point in said PWM generation table upon generation of a timer overflow interrupt,

wherein said step of generating an interrupt further comprises the step of writing values to a plurality of port pins directly from said PWM generation table.

14. (New) A method for generating multiple high-resolution pulse width modulated (PWM) signals comprising the steps of:

receiving data representative of duty cycle values;

sorting said duty cycle values in a duty cycle table to generate a PWM generation table;

generating an interrupt from a capture and compare interrupt defined by a match between a timer value and a duty cycle register;

generating an interrupt from a timer overflow;

generating multiple PWM signals as defined by said PWM generation table upon detection of said capture and compare interrupt; and

returning to a beginning point in said PWM generation table upon generation of a timer overflow interrupt,

wherein said step of generating an interrupt from a timer overflow further comprises the step of resetting a plurality of port pins to low.

15. (New) The method as claimed in claim 12 wherein said step of sorting said duty cycle values is done in a background task.

16. (New) The method as claimed in claim 13 wherein said step of sorting said duty cycle values is done in a background task.

17. (New) The method as claimed in claim 14 wherein said step of sorting said duty cycle values is done in a background task.